Comp. & Maths. with Appls. Vol. 12A, No. 9, pp. 1007–1017, 1986
Printed in Great Britain.
0886-9553/86 $3.00 + 0.00

RUNGE–KUTTA TRIPLES

J. R. DORMAND[1] and P. J. PRINCE[2]
Departments of [1]Mathematics and Statistics and [2]Computer Science, Teesside Polytechnic, Middlesbrough, Cleveland TS1 3BA, England

(*Received February 1986*)

Communicated by L. F. Shampine

Abstract—The addition of a third Runge–Kutta (RK) formula to an embedded pair, forming an RK triple, can yield an algorithm which gives solutions of appropriate asymptotic accuracy at points within the normal step intervals in the numerical solution of the initial-value problem. Two modes of implementation are possible, and it is shown that these can be regarded as equivalent when certain conditions are satisfied. An optimized RK5(4) triple, based on a particularly efficient embedded pair, is presented and tested. The results indicate the efficiency of the "dense" output technique.

1. INTRODUCTION

We consider the first-order system of non-stiff ordinary differential equations:

$$\mathbf{y}'(x) = \mathbf{f}[x, \mathbf{y}(x)], \quad \text{with} \quad \mathbf{y}(x_0) \text{ known.}$$

The system may be solved using an embedded Runge–Kutta (RK) process with formulae of orders q and $p (q > p)$ of the form:

$$\hat{\mathbf{y}}_{n+1} = \hat{\mathbf{y}}_n + h_n \hat{\boldsymbol{\Phi}}(x_n, \hat{\mathbf{y}}_n, h_n)$$

and

$$\mathbf{y}_{n+1} = \hat{\mathbf{y}}_n + h_n \boldsymbol{\Phi}(x_n, \hat{\mathbf{y}}_n, h_n) \tag{1}$$

where

$$\hat{\boldsymbol{\Phi}}(x_n, \hat{\mathbf{y}}_n, h_n) = \sum_{i=1}^{s} \hat{b}_i \mathbf{g}_i,$$

$$\boldsymbol{\Phi}(x_n, \hat{\mathbf{y}}_n, h_n) = \sum_{i=1}^{s} b_i \mathbf{g}_i$$

and

$$\mathbf{g}_i = \mathbf{f}(x_n + c_i h_n, \hat{\mathbf{y}}_n + h_n \sum_{j=1}^{i-1} a_{ij} \mathbf{q}_j), \quad i = 1, 2, \ldots, s \tag{2}$$

The latter summation is taken as zero when

$$i = 1, \quad c_i = \sum_{j=1}^{i-1} a_{ij}, \quad x_{n+1} = x_n + h_n, \quad h_n = \theta(x_n) h, \quad 0 < \theta(x) \leqslant 1, \quad \hat{\mathbf{y}}_0 = \mathbf{y}(x_0)$$

and $\hat{\mathbf{y}}_n$ and $\mathbf{y}_n$ denote the approximations to the true solution $\mathbf{y}(x_n)$ from the formulae of orders q and p, respectively. It should be noted that the embedded process is applied here in local extrapolation, or higher-order, mode [1], i.e. the qth order approximation $\hat{\mathbf{y}}_n$ is used as the initial value for the $(n+1)$th step. Assuming appropriate smoothness of $\mathbf{f}$, the local truncation error, $\hat{\mathbf{t}}_{n+1}$, at x_{n+1} of the RKq process may be written [2] as

$$\hat{\mathbf{t}}_{n+1} = \mathbf{y}(x_n) + h_n \hat{\boldsymbol{\Phi}}[x_n, \mathbf{y}(x_n), h_n] - \mathbf{y}(x_{n+1}) = \sum_{i=q}^{\infty} h_n^{i+1} \hat{\boldsymbol{\phi}}_i[x_n, \mathbf{y}(x_n)], \tag{3}$$

where

$$\hat{\boldsymbol{\phi}}_i[x, \mathbf{y}(x)] = \sum_{j=1}^{n_{i+1}} \hat{\tau}_j^{(i+1)} \mathbf{F}_j^{(i-1)}[x, \mathbf{y}(x)], \quad i = 0, 1, \ldots$$

A similar expression holds for $\mathbf{t}_{n+1}$, the local truncation error of the RKp. The error coefficients (τ) are functions of the RK parameters (a, b, c) [3], which are chosen so that

$$\left.\begin{aligned} \hat{\tau}_j^{(i+1)} &= 0, \quad i = 0, 1, \ldots, q-1 \\ \tau_j^{(i+1)} &= 0, \quad i = 0, 1, \ldots, p-1 \end{aligned}\right\} j = 1, 2, \ldots, n_{i+1}.$$

The number of stages, s (the number of times $\mathbf{f}$ is evaluated at each step of the integration), is usually kept as small as possible. Some embedded formulae make use of the FSAL idea [4] in which the last evaluation at any step is the same as the first at the next step. For a formula employed in local extrapolation mode this requires

$$c_s = 1, \quad \hat{b}_s = 0 \quad \text{and} \quad a_{sj} = \hat{b}_j, \quad j = 1, 2, \ldots, s-1. \tag{4}$$

Since $\hat{b}_s = 0$ then $\mathbf{g}_s$ makes no contribution to $\hat{\mathbf{y}}_{n+1}$. Normally $b_s \neq 0$ and so $\mathbf{g}_s$ is used in error estimation at the current step. The case where $b_s = 0$ can be considered equivalent to the non-FSAL formula using $(s-1)$ stages but the calculation of $\mathbf{g}_s$ would not be wasted since normally it is required to start the next integration step. In either case, for steps which are not rejected (after the first step), $(s-1)$ new function evaluations must be computed. Bearing this in mind it will be seen that there is no loss of generality in the assumption of FSAL in what follows.

2. RK TRIPLES

Recently Horn [5], Shampine [6, 7], Enright *et al.* [8] and Gladwell *et al.* [9] have considered the development of RK processes which produce numerical solutions for $\mathbf{y}(x)$ at non-mesh points, a capability which enhances the practicality of any variable step-length integrator. One way of achieving this is to add a third RK formula to the embedded pair (called a *dense* formula) which will be used to integrate from x_n with a step of size σh_n, where normally $0 < \sigma < 1$.

The dense formula may be expressed in the usual manner:

$$\mathbf{y}^*_{n+\sigma} = \hat{\mathbf{y}}_n + \sigma h_n \boldsymbol{\Phi}^*(x_n, \hat{\mathbf{y}}_n, \sigma h_n) \tag{5}$$

where

$$\boldsymbol{\Phi}^*(x_n, \hat{\mathbf{y}}_n, \sigma h_n) = \sum_{i=1}^{s^*} b_i^* \mathbf{g}_i^*$$

and

$$\mathbf{g}_i^* = \mathbf{f}\left(x_n + c_i^* \sigma h_n, \hat{\mathbf{y}}_n + \sigma h_n \sum_{j=1}^{i-1} a_{ij}^* \mathbf{g}_j^*\right), \quad i = 1, 2, \ldots, s^*$$

Ideally we would like $s^* = s$ and $\mathbf{g}_i^* = \mathbf{g}_i, i = 1, 2, \ldots, s^*$, so that no extra function evaluations are necessary to compute the approximation for $\mathbf{y}(x_n + \sigma h_n)$. This may or may not be possible depending on q, p and p^*, the order of the dense formula. In order to satisfy the RKp^* equations of condition it may be necessary that $s^* > s$ so that extra RK parameters are available. If the following relations are satisfied then common function evaluations ($\mathbf{g}_i$) are guaranteed:

$$\begin{aligned} c_i^* &= c_i/\sigma, \quad i = 1, 2, \ldots, s_{\mathrm{m}}, \\ a_{ij}^* &= a_{ij}/\sigma, \quad i = 2, 3, \ldots, s_{\mathrm{m}}, \quad j = 1, 2, \ldots, i-1, \end{aligned} \tag{6}$$

and

$$s_{\mathrm{m}} = \max(s, s^*).$$

It is interesting to note the form of the RK truncation error coefficients for the "dense" formula; the first few may be written as

$$\tau_1^{(1)^*} = \sum_i b_i^* - 1,$$

$$\sigma\tau_1^{(2)^*} = \sum_i b_i^* c_i - \frac{\sigma}{2},$$

$$\sigma^2\tau_1^{(3)^*} = \tfrac{1}{2}\sum_i b_i^* c_i^2 - \frac{\sigma^2}{6} \tag{7}$$

and

$$\sigma^2\tau_2^{(3)^*} = \sum_{ij} b_i^* a_{ij} c_j - \frac{\sigma^2}{6},$$

where all the summations run from 1 to s^*. Since the number of available degrees of freedom is small for embedded RK pairs, it is possible to set $s^* = s$ only for low-to-moderate values of p^*. The most obvious choice for p^* is either p or q, and so for high-order formulae (say $p, q \geqslant 5$) it is likely that we must have $s^* > s$. The embedded pair together with the "dense" formula constitute an RK triple using s_m stages such that if $s_m > s$, then $\hat{b}_i = b_i = 0$, $i = s+1, s+2, \ldots, s_m$. It is clear that the b_i^* are functions of σ.

There are two methods of implementing dense output solutions:

(A) if the RKp^* equations can be satisfied for all $\sigma \in [0, 1]$, the $b_i^*(\sigma)$ may be computed for any desired output point and hence equation (5) may be applied directly;

(B) an interpolating polynomial based on the end points and some intermediate points (possibly including derivatives) obtained from equation (5) can be constructed for any step interval.

The second method is necessary if the RKp^* is available only for specific values of σ. For example [5], Fehlberg's RKF4 pair permits a one-parameter family of RK4* with $\sigma = 3/5$ and $s^* = s$. Enright *et al.* [8] optimize the choice of the parameter in a certain sense for use in mode B. The RK5(4)7FM [1] allows RK4* for any σ with $s^* = s$. Shampine [6] has obtained this RK4* for $\sigma = 1/2$, and has implemented this formula in mode B, although it can also be used in mode A. The precise relations between methods A and B will be considered later in this paper.

It is necessary to justify any choice for p^*. We would argue that an estimate of $\mathbf{y}(x_n + \sigma h_n)$ should have the same global order of accuracy as $\hat{\mathbf{y}}_n$. Now the global error at a main solution point is

$$\hat{\boldsymbol{\epsilon}}_n = \hat{\mathbf{y}}_n - \mathbf{y}(x_n) = O(h^q), \quad n = 0, 1, 2, \ldots$$

and so we require

$$\boldsymbol{\epsilon}^*_{n+\sigma} = \mathbf{y}^*_{n+\sigma} - \mathbf{y}(x_n + \sigma h_n) = O(h^q) \tag{8}$$

for mode A, and

$$\mathbf{P}(x_n + \sigma h_n) - \mathbf{y}(x_n + \sigma h_n) = O(h^q) \tag{9}$$

for mode B, where $\mathbf{P}$ is the interpolating polynomial. We now consider the two modes separately.

(i) Mode A

Using equations (5) and (8) together with the definition of the local truncation error for the RKp^* formula [similar to equation (3)], we find

$$\boldsymbol{\epsilon}^*_{n+\sigma} = \hat{\boldsymbol{\epsilon}}_n + \sigma h_n\{\boldsymbol{\Phi}^*[x_n, \mathbf{y}_n, \sigma h_n] - \boldsymbol{\Phi}^*[x_n, \mathbf{y}(x_n), \sigma h_n]\} - \mathbf{t}^*_{n+\sigma},$$

which, assuming Φ^* satisfies a Lipschitz condition, gives

$$\| \epsilon^*_{n+\sigma} \| \leqslant \| \hat{\epsilon}_n \| + \sigma h_n L \| \hat{\epsilon}_n \| + \| \mathbf{t}^*_{n+\sigma} \|. \tag{10}$$

Now $\| \hat{\epsilon}_n \|$ is $O(h^q)$ and $\| \mathbf{t}^*_{n+\sigma} \|$ is $O(h^{p^*+1})$ and so for $\| \epsilon^*_{n+\sigma} \|$ to be $O(h^q)$ we require $p^* \geqslant q-1$. Since we do not wish to satisfy any more equations of condition than necessary, bearing in mind the lack of degrees of freedom, we shall take $p^* = q-1$. For some applications [8, 9] it is preferable to choose $p^* = q$.

With regard to the global continuity of the RK interpolation we will have C^1 continuity if

$$\mathbf{y}^*_{n+\sigma}(\sigma = 0) = \hat{\mathbf{y}}_n, \quad \mathbf{f}(x_{n+\sigma}, \mathbf{y}^*_{n+\sigma})(\sigma = 0) = \mathbf{f}(x_n, \hat{\mathbf{y}}_n),$$

$$\mathbf{y}^*_{n+\sigma}(\sigma = 1) = \hat{\mathbf{y}}_{n+1} \quad \text{and} \quad \mathbf{f}(x_{n+\sigma}, \mathbf{y}^*_{n+\sigma})(\sigma = 1) = \mathbf{f}(x_{n+1}, \hat{\mathbf{y}}_{n+1}).$$

The first two conditions are satisfied [equation (5) with $\sigma = 0$] and the latter two will be true if [see equation (5)]

$$b^*_i(\sigma = 1) = \hat{b}_i, \quad i = 1, 2, \ldots, s_m$$

This will be discussed again in Section 4 of this paper.

(ii) Mode B

Let $\mathbf{P}_m(x)$ be the polynomial of degree $\leqslant m$ which interpolates the $m+1$ points

$$(x_{n+\sigma_k}, \mathbf{y}^*_{n+\sigma_k}), \quad k = 1, 2, \ldots, r_1$$

and

$$(x_{n+\sigma_k}, \mathbf{f}[x_{n+\sigma_k}, \mathbf{y}^*_{n+\sigma_k}]), \quad k = 1, 2, \ldots, r_2, \quad r_1 \geqslant r_2 \geqslant 0, \quad r_1 \geqslant 3,$$

where $m = r_1 + r_2 - 1$, $\sigma_1 = 0$, $\sigma_2 = 1$, $\mathbf{y}^*_{n+\sigma_1} = \hat{\mathbf{y}}_n$ [from equation (5)] and $\mathbf{y}^*_{n+\sigma_2}$ must be replaced by $\hat{\mathbf{y}}_{n+1}$ wherever it occurs. Note that $\mathbf{y}^*_{n+\sigma_2}$ is not necessarily equal to $\hat{\mathbf{y}}_{n+1}$ (see mode A subsection).

These latter conditions involving σ_1 and σ_2 imply that the interpolant is globally C^0. Since we are assuming FSAL (Section 1), it will be noted that

$$\mathbf{f}(x_{n+\sigma_1}, \mathbf{y}^*_{n+\sigma_1}) = \mathbf{f}(x_n, \hat{\mathbf{y}}_n) = \mathbf{g}_1$$

and (11)

$$\mathbf{f}(x_{n+\sigma_2}, \mathbf{y}^*_{n+\sigma_2}) \quad \text{is replaced by} \quad \mathbf{f}(x_{n+1}, \hat{\mathbf{y}}_{n+1}) = \mathbf{g}_s$$

Note that equations (11) imply that the polynomial interpolant is globally C^1, as indicated by Shampine [7]. However, Horn's interpolant [5] is only C^0. We take $r_2 = 2$ so that no extra function evaluations are necessary as a result of the interpolation, but Gladwell *et al.* [9] prefer $r_2 = 3$ for their application to a 5(4) pair. The interpolation will be a modified Hermite [10] unless $r_2 = 0$, which implies a Lagrangian interpolant based only on y-values. Let $\mathbf{V}_m(x)$ be the polynomial of degree $\leqslant m$ which interpolates the true solution and derivatives at the same points. Then

$$\mathbf{P}_m(x) = \sum_{k=1}^{r_1} L_k(x) \mathbf{y}^*_{n+\sigma_k} + \sum_{k=1}^{r_2} M_k(x) \mathbf{f}(x_{n+\sigma_k}, \mathbf{y}^*_{n+\sigma_k}) \tag{12}$$

and

$$\mathbf{V}_m(x) = \sum_{k=1}^{r_1} L_k(x) \mathbf{y}(x_{n+\sigma_k}) + \sum_{k=1}^{r_2} M_k(x) \mathbf{f}[x_{n+\sigma_k}, \mathbf{y}(x_{n+\sigma_k})] \tag{13}$$

Subtracting, and assuming $\mathbf{f}$ is sufficiently smooth, gives

$$\| \mathbf{P}_m(x) - \mathbf{V}_m(x) \| \leqslant \sum_{k=1}^{r_1} \| L_k(x) \epsilon^*_{n+\sigma_k} \| + \sum_{k=1}^{r_2} \| M_k(x) \mathrm{A}_k \epsilon^*_{n+\sigma_k} \|,$$

where the A_k are Lipschitz constants. Now the Lagrange–Hermite polynomials, L_k and M_k, are $O(1)$ and $O(h)$, respectively [3, 7], so that the order of $\| \mathbf{P}_m - \mathbf{V}_m \|$ is dependent on the

order of $\|\epsilon^*_{n+\sigma_k}\|$ at the σ_k points. For $k = 1, 2$, $\|\epsilon^*_{n+\sigma_k}\| = O(h^q)$, and for $k > 2$, similar to inequality (10) we have

$$\|\epsilon^*_{n+\sigma_k}\| \leqslant \|\hat{\epsilon}_n\| + \sigma_k h_n L \|\hat{\epsilon}_n\| + \|\mathbf{t}^*_{n+\sigma_k}\|$$

Thus, provided the minimum order of the RKp^* at the σ_k points is $q-1$ then $\|\epsilon^*_{n+\sigma_k}\|$, and hence $\|\mathbf{P}_m(x) - \mathbf{V}_m(x)\|$, is $O(h^q)$.

Now $\mathbf{V}_m(x)$ is of degree $\leqslant m$, and so

$$\|\mathbf{V}_m(x) - \mathbf{y}(x)\| \leqslant \mathrm{B}h^{m+1}, \quad x \in [x_n, x_{n+1}]$$

where B is a constant, giving

$$\|\mathbf{P}_m(x) - \mathbf{y}(x)\| \leqslant \mathrm{C}h^{\min[q, m+1]}.$$

Thus, provided $m \geqslant q-1$, $\mathbf{P}_m(x)$ may be used to obtain an approximation of the correct order.

3. EQUIVALENCE OF MODES A AND B

It is instructive to consider the equivalence of the two cases, A and B. Shampine [7] has shown that interpolants in the form of mode B can be expressed in mode-A form and also that Horn's fourth-order mode-A interpolant [5] can be written as a quartic polynomial. There now follows a more general analysis which will show that the two modes are exactly equivalent under certain conditions. Consider again $\mathbf{V}_m(x)$ and let $x = x_n + \sigma h_n$. Then from equation (13)

$$\mathbf{V}_m(\sigma) = \sum_{k=1}^{r_1} L_k(\sigma)\mathbf{y}(x_{n+\sigma_k}) + \sum_{k=1}^{r_2} M_k(\sigma)\mathbf{f}[x_{n+\sigma_k}, \mathbf{y}(x_{n+\sigma_k})]$$

and the interpolation error is given by

$$\mathbf{E}_m(\sigma) = h_n^{r_1+r_2} \prod_{i=1}^{r_1} (\sigma - \sigma_i) \prod_{j=1}^{r_2} (\sigma - \sigma_j)\mathbf{y}^{(r_1+r_2)}(\zeta)/(r_1+r_2)!,$$

where the second product will be taken as unity in the Lagrange case where $r_2 = 0$. Thus $\mathbf{V}_m$ is exact for all polynomials of degree $< r_1 + r_2$ and, hence,

$$\sum_{k=1}^{r_1} L_k(\sigma)\sigma_k^t + \sum_{k=1}^{r_2} t\sigma_k^{t-1} M(\sigma)/h_n = \sigma^t, \quad t = 0, 1, \ldots, r_1 + r_2 - 1. \tag{14}$$

Using equations (5) and (12),

$$\mathbf{P}_m(\sigma) = \sum_{k=1}^{r_1} L_k(\sigma)\left\{\hat{\mathbf{y}}_n + \sigma_k h_n \sum_{i=1}^{s_m} b_i^*(\sigma_k)\mathbf{g}_i\right\} + \sum_{k=1}^{r_2} M_k(\sigma)\mathbf{f}(\sigma_k). \tag{15}$$

where $\mathbf{f}(\sigma_k) = \mathbf{f}(x_{n+\sigma_k}, \mathbf{y}^*_{n+\sigma_k})$ and, with $k = 2$, $b_i^*(\sigma_k)$ is replaced by b_i, $i = 1, 2, \ldots, s_m$.

Using equation (14) with $t = 0$ now gives

$$\mathbf{P}_m(\sigma) = \hat{\mathbf{y}}_n + \sigma h_n \left\{\sum_{i=1}^{s_m}\left[\sum_{k=1}^{r_1} \sigma_k L_k(\sigma) b_i^*(\sigma_k)/\sigma\right]\mathbf{g}_i + \sum_{k=1}^{r_2} M_k(\sigma)\mathbf{f}(\sigma_k)/(\sigma h_n)\right\}.$$

This is of the form of a dense RK formula,

$$\mathbf{y}^*_{n+\sigma} = \hat{\mathbf{y}}_n + \sigma h_n \sum_{i=1}^{\nu} B_i^*(\sigma)\mathbf{g}_i$$

in $\nu = s_m + r_2 - 2$ stages, where [using equations (11)]

$$B_1^*(\sigma) = \left[\sum_{k=1}^{r_1} \sigma_k L_k(\sigma) b_1^*(\sigma_k) + M_1(\sigma)/h_n\right]\Big/\sigma,$$

$$B_i^*(\sigma) = \sum_{k=1}^{r_1} \sigma_k L_k(\sigma) b_i^*(\sigma_k)/\sigma, \quad i = 2, 3, \ldots, s-1, \quad s+1, \ldots, s_m,$$

$$B_s^*(\sigma) = \left[\sum_{k=1}^{r_1} \sigma_k L_k(\sigma) b_s^*(\sigma_k) + M_2(\sigma)/h_n\right]\Big/\sigma \tag{16}$$

and

$$B_i^*(\sigma) = \left[\sum_{k=1}^{r_1} \sigma_k L_k(\sigma) b_i^*(\sigma_k) + M_{i-s_m+2}(\sigma)/h_n\right] \Big/ \sigma, \quad i = s_m + 1, \ldots, v,$$

where we have taken $b_i^*(\sigma_k) = \hat{b}_i = 0, \quad i = s_m + 1, \ldots, v$ and

$$\mathbf{g}_{s_m+k-2} = \mathbf{f}(\sigma_k) = \mathbf{f}\left(x_n + \sigma_k h_n, \hat{\mathbf{y}}_n + \sigma_k h_n \sum_{j=1}^{s_m+k-3} b_{s_m+k-2}^*(\sigma_k)\mathbf{g}_j\right), \quad k = 3, 4, \ldots, r_2,$$

which can be regarded as RK function evaluations of the form given in equations (2) where

$$c_{s_m+k-2} = \sigma_k, \quad k = 3, 4, \ldots, r_2, \quad j = 1, 2, \ldots, k-1 \tag{17}$$

and

$$a_{s_m+k-2,j} = \sigma_k b_{s_m+k-2}^*(\sigma_k).$$

Now the $b_i^*(\sigma_k)$ ($\hat{b}_i$ when $k = 2$) satisfy the RK equations of condition [3], which may be written in the form [11]

$$\Psi = [\Psi_1, \Psi_2, \ldots, \Psi_d] = \sum_{i=1}^{v} b_i^*(\sigma_k) R_{iu}^{(t)} = \sigma_k^{t-1}/\gamma_u^{(t)}, \tag{18}$$

$$u = 1, 2, \ldots, n_t, \quad t = 1, 2, \ldots, p_m = \min(p^*, q),$$

where

$$R_{iu}^{(t)} = \begin{cases} 1, & t = 1 \\ \displaystyle\sum_{j_1, \ldots, j_d = 1}^{v} a_{ij_1} a_{ij_2} \ldots a_{ij_d} \Psi_1^{(j_1)} \Psi_2^{(j_2)} \ldots \Psi_d^{(j_d)}, & t > 1, \end{cases} \tag{19}$$

where

$$\Psi_j = \sum_{i=1}^{v} b_i^*(\sigma_k) \Psi_j^{(i)}, \quad \Psi_j^{(i)}$$

being independent of b_i^* and thus so is $R_{iu}^{(t)}$. Taking equation (16), multiplying by the appropriate $R_{iu}^{(t)}$ and summing over i from 1 to v gives

$$\sum_{i=1}^{v} B_i^*(\sigma) R_{iu}^{(t)} = \sum_{k=1}^{r_1} \sigma_k L_k(\sigma)/\sigma \sum_{i=1}^{v} b_i^*(\sigma_k) R_{iu}^{(t)} + M_1(\sigma)/(\sigma h_n) R_{1u}^{(t)}$$
$$+ M_2(\sigma)/(\sigma h_n) R_{su}^{(t)} + \sum_{i=s_m+1}^{v} M_{i-s_m+2}(\sigma)/(\sigma h_n) R_{iu}^{(t)},$$

which using equations (18) and (14) gives

$$\sum_{i=1}^{v} B_i^*(\sigma) R_{iu}^{(t)} = \sigma^{t-1}/\gamma_u^{(t)} + M_1(\sigma)/(\sigma h_n)[R_{1u}^{(t)} - t\sigma_1^{t-1}/\gamma_u^{(t)}]$$
$$+ M_2(\sigma)/(\sigma h_n)[R_{su}^{(t)} - t\sigma_2^{t-1}/\gamma_u^{(t)}]$$
$$+ \sum_{k=3}^{r_2} M_k(\sigma)/(\sigma h_n)[R_{(k+s_m-2)u}^{(t)} - t\sigma_k^{t-1}/\gamma_u^{(t)}], \tag{20}$$
$$u = 1, 2, \ldots, n_t, \quad t = 1, 2, \ldots, p_m, \quad r_1 + r_2 - 1 \geqslant p_m.$$

Now, because $\sigma_1 = 0$ and $a_{1j} = 0$ for an explicit RK process then $R_{1u}^{(t)}$ and σ_1^{t-1} are zero unless $t = 1$ and when $t = 1$, then $n_1 = 1$, $\gamma_u^{(t)} = 1$ and $R_{iu}^{(t)} = 1$ from equation (19) and so the coefficient of $M_1(\sigma)$ in equation (20) is zero. Thus using equations (4), (15) and (19),

$$R_{su}^{(t)} = \sum_{j_1 \ldots j_d = 1}^{v} \hat{b}_{j_1} \Psi_1^{(j_1)} \hat{b}_{j_2} \Psi_2^{(j_2)} \ldots \hat{b}_{j_d} \Psi_d^{(j_d)} = \frac{1}{\gamma_1} \cdot \frac{1}{\gamma_2} \cdots \frac{1}{\gamma_d}$$

since each of

$$\sum_{j_i=1}^{v} \hat{b}_{j_i} \Psi_i^{(j_i)}$$

is an RK equation of order $<t$. So, using result (31) of Ref. [9],

$$R_{iu}^{(t)} = t/\gamma_u^{(t)}$$

and so, since $\sigma_2 = 1$, the coefficient of $M_2(\sigma)$ in equation (20) is also zero. In a similar manner using equations (17) and (19) it may be shown that the coefficient of $M_k(\sigma)$, $k = 3, 4, \ldots, r_2$ is zero. Thus equation (20) now gives

$$\sum_{i=1}^{v} B_i^*(\sigma) R_{iu}^{(t)} = \sigma^{t-1}/\gamma_u^{(t)}, \quad u = 1, 2, \ldots, n_t, \quad t = 1, 2, \ldots, p_m, \quad r_1 + r_2 - 1 \geqslant p_m.$$

Thus provided $r_1 + r_2 - 1 \geqslant p_m$, the polynomial interpolation approach B is equivalent to an $(s_m + r_2 - 2)$ stage RKp_m formula. Therefore since our normal requirement from Section 2 is $p^* = q - 1$, for equivalence we must have $r_1 + r_2 \geqslant q$.

4. AN RK5(4) TRIPLE

We now illustrate the dense output technique by deriving an RK4* based on the RK5(4) theory of Ref. [1], where $s = 7$ and FSAL was employed. The RK4* must satisfy eight equations of condition and for no extra function evaluations we require $s_m = 7$ (and $r_2 \leqslant 2$ if mode B is used). It might seem, therefore, that in this case because we have eight extra equations and only seven extra RK parameters, b_i^*, $i = 1, 2, \ldots, 7$, an additional constraint would be placed on the RK5(4) model of Ref. [1]. This is not so since in this model the following simplifying assumptions were made:

$$\sum_{j=1}^{s} a_{ij} c_j^k = \frac{c_i^{k+1}}{k+1}, \quad i = 3, 4, \ldots, s, \quad k = 1, 2.$$

Thus taking $b_2^* = 0$ the eight equations reduce to the following five independent ones [see equation (7)]:

$$\sum_{i=1}^{7} b_i^* = 1 \tag{21}$$

$$\sum_{i=3}^{7} b_i^* c_i^k = \sigma^k/(k+1), \quad k = 1, 2, 3 \tag{22}$$

and

$$\sum_{i=3}^{7} b_i^* a_{i2} = 0. \tag{23}$$

These may be solved for general σ by treating b_7^* as a free parameter, using equations (22) and (23) as linear equations to solve for b_i^*, $i = 3, 4, 5, 6$, and then obtaining b_1^* from equation (21). Since no additional constraints have been imposed on the RK5(4) model as a result of the RK4*, the RK5(4)7FM of Ref. [1] may be used as the embedded pair. Prince and Dormand [12] adopted the strategy of minimizing the principal truncation error coefficients in deriving near-"optimum" formulae and so, following Shampine [6], it is natural to choose b_7^* to minimize

$$\| \tau^{(5)*} \| = \sqrt{\sum_{j=1}^{9} [\tau_j^{(5)*}]^2}. \tag{24}$$

Since

$$\tau_j^{(5)*} = \zeta_j(\sigma) + \rho_j(\sigma) b_7^*, \quad j = 1, \ldots, 9$$

equation (24) will be minimized if

$$b_7^* = -\sum_{j=1}^{9} \zeta_j \rho_j \Big/ \sum_{j=1}^{9} \rho_j^2,$$

yielding

$$b_7^* = \frac{\sigma(1-\sigma)(8293050\sigma^2 - 82437520\sigma + 44764047)}{29380423}$$

and, hence,

$$b_1^* = (157015080\sigma^4 - 13107642775\sigma^3 + 34969693132\sigma^2 - 32272833064\sigma + 11282082432)/11282082432,$$

$$b_2^* = 0,$$

$$b_3^* = -100\sigma(15701508\sigma^3 - 914128567\sigma^2 + 2074956840\sigma - 1323431896)/32700410799,$$

$$b_4^* = 25\sigma(94209048\sigma^3 - 1518414297\sigma^2 + 2460397220\sigma - 889289856)/5641041216,$$

$$b_5^* = -2187\sigma(52338360\sigma^3 - 451824525\sigma^2 + 687873124\sigma - 259006536)/199316789632$$

and

$$b_6^* = 11\sigma(106151040\sigma^3 - 661884105\sigma^2 + 946554244\sigma - 361440756)/2467955532.$$

Thus these coefficients define the "optimum" RK4*. It will be seen that setting $\sigma = 1$ in $b_i^*(\sigma)$ yields the $\hat{b}_i$, indicating C^1 continuity of the global RK interpolant. Since $\hat{b}_7 = 0$ and b_7^* is free, an arbitrary choice of b_7^* will not generally give even C^0 global continuity for the mode-A RK interpolant. However the minimization of $\| \tau^{(5)*} \|$ yields $b_i^*(\sigma = 1) = \hat{b}_i$, $i = 1, 2, \ldots, 7$. This is not too surprising because of the similarity of the equations being satisfied by the two formulae. Setting $\sigma = 1/2$ gives the same values as quoted by Shampine [6] for the DPS triple which would be used in mode B with $r_2 = 2$. Shampine minimized the error coefficients with $\sigma = 1/2$ because of his preference for mode B. As shown in Section 3, the interpolating polynomial in the DPS triple is equivalent to an RK4* formula since $m = 4$. This equivalent RK4* is not the "optimum" one specified above unless $\sigma = 1/2$. The difference, however, is not significant.

5. NUMERICAL TESTS

To test the efficiency of the dense output technique we have applied the RK5(4)7FM triple in mode A to the gravitational two-body problem

$$\mathbf{y}''(x) = -\mathbf{y}(x)/r^3,$$

where $r^2 = |\mathbf{y}|^2$, with initial conditions

$$\mathbf{y}(0) = \begin{bmatrix} 1-e \\ 0 \end{bmatrix}, \quad \mathbf{y}'(0) = \begin{bmatrix} 0 \\ \sqrt{(1+e)/(1-e)} \end{bmatrix}.$$

This problem is designated D in the well-known DETEST [13] set of test problems. It becomes more difficult as e increases [$e \in (0, 1)$] because of the large variation in the magnitude of $\mathbf{f}$. We present global error data for D1 ($e = 0.1$), D3 ($e = 0.5$) and D5 ($e = 0.9$). The dense formula was employed first to give solutions at $x = 1$ to 20 in steps of 1 and the maximum global error (over all four components and steps) is compared with that of the main solution points computed from the RK5. The tolerance was varied between 10^{-3} and 10^{-9} and absolute error per unit step was used. The results are presented in Table 1 and for D5 in particular, Fig. 1 shows the efficiency curves. For tolerances other than 10^{-3}, where the global error is greater than the true solution with problem D1, it will be seen that the maximum global error for the dense solution is lower than that for the normal solution. This is particularly noticeable in the case of D5 and requires some explanation. The global errors of the first y-component in the D5 case are plotted in Fig. 2. It is clear that our selection of output points does not include values very close to the positions of extreme global error which occur near multiples of 2π. Thus it is not surprising

Table 1. Maximum global errors at normal solution points (N) and at dense output points (D)

Log_{10} (tolerance)	Problem D1		Problem D3		Problem D5	
	N	D	N	D	N	D
−3	0.20	0.21	−0.55	−0.52	0.33	−0.93
−4	−0.86	−0.87	−2.08	−2.15	−0.87	−2.04
−5	−2.51	−2.51	−3.01	−3.08	−2.60	−3.80
−6	−4.42	−4.42	−4.37	−4.46	−3.04	−4.24
−7	−6.05	−6.05	−5.96	−6.21	−4.08	−5.28
−8	−6.95	−6.95	−7.25	−7.31	−5.16	−6.36
−9	−8.13	−8.13	−8.58	−8.63	−6.19	−7.39

Tabular values are logarithms (base 10) of maximum global errors over all variables and steps.

that we seem to be achieving smaller errors with the dense formula. The extreme-point errors are relatively much smaller with problem D1. Changing the range of dense output solution to [18, 19] within which the maximum global error occurs (see Fig. 1), results in a situation in which the dense points have the same maximum error (to three significant figures) as the normal solution points. Obviously, for lax tolerances ($\geqslant 10^{-3}$) poorer results are to be expected because of the asymptotic nature of the analysis.

It is important to note that use of preselected dense output points *only* as a representation of the solution of a system of equations may be misleading. This would

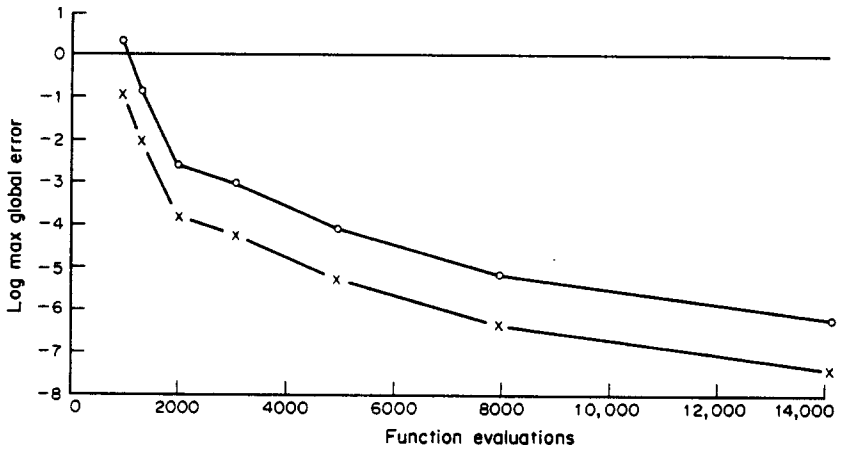

Fig. 1. Log_{10}|maximum global error (all steps and variables)| against function evaluations for problem D5: O, normal variable-step solution; ×, dense output solution ($x = 1$ to 20 in steps of 1).

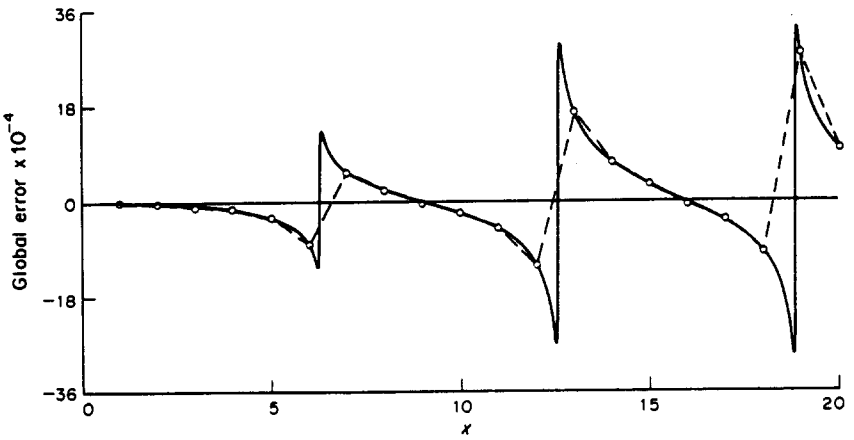

Fig. 2. Actual global error of problem D5 with tolerance 10^{-4} against x (——) and global errors at dense output points (---).

certainly be the case with D5 (Fig. 1). The embedded variable-step algorithm will, for sufficiently stringent tolerances, compute points near to values of x where $\mathbf{f}$ is varying most rapidly and thus will most likely give a good representation of the solution over the required range.

Although, in general, the dense formula will yield solutions comparable in accuracy (see Section 2) to those of the normal integrator it will be found, in some cases, that the dense solution has a larger absolute global error than at the neighbouring mesh points. Within the first step, the RKp^* is of lower order globally than the RKq.

All tests quoted here have been carried out using mode A. Tests carried out using mode B, with $p^* = q - 1$, gave results which were not significantly different; this is not surprising in view of the equivalence of the modes when $r_1 + r_2 - 1 \geqslant p^*$. Tests have also been conducted with non-optimized RKp^* using mode A; the errors for dense output were generally larger.

6. DISCUSSION

The above consideration of the dense output technique allied with embedded RK methods to form triples has indicated its practical value. Although two modes of implementation are possible we prefer mode A in which a third RK formula is used directly. Horn [5] and Enright *et al.* [8] have the same preference but Shampine [7] and Gladwell *et al.* [9] prefer mode B. Differing user requirements might make either of the modes computationally more efficient. For example, if derivative estimations are required then mode A requires the evaluation of $\mathbf{f}(x_{n+\sigma}, \mathbf{y}^*_{n+\sigma})$ for each one, whereas in mode B numerical differentiation would suffice, provided the interpolant has high enough degree. In the latter case no extra function evaluations are required once the interpolant has been formed. Numerical tests have indicated that as in Ref. [12] the formula optimized with respect to local truncation error coefficients is generally to be preferred. Also the quality of RK interpolated solutions is very similar to that of the normal variable-step solution. The same technique is applicable to higher-order embedded RK pairs but in view of the extra constraints it may be necessary to add function evaluations to the dense output formula, thus increasing cost. This problem is under active consideration. A similar analysis of dense output techniques for Runge–Kutta–Nystrom (RKN) processes applied to the special second-order initial-value problem

$$\mathbf{y}''(x) = \mathbf{f}[x, \mathbf{y}(x)], \quad \mathbf{y}(x_0), \quad \mathbf{y}'(x_0) \quad \text{known},$$

is currently being examined.

The application of the technique to global error estimation using the Zadunaisky process [2, 3] is also being considered. It will be clear that a defect function based on dense points within a single integration step can be constructed. Initial tests indicate valid error estimation is possible under certain conditions.

REFERENCES

1. J. R. Dormand and P. J. Prince, A family of embedded Runge-Kutta formulae. *J. Comput. appl. Math.* **6**(1), 19–26 (1980).
2. J. R. Dormand and P. J. Prince, Global error estimation with Runge–Kutta methods II. *IMA Jl numer. Analysis* **5**, 481–497 (1985).
3. J. R. Dormand, R. R. Duckers and P. J. Prince, Global error estimation with Runge–Kutta methods. *IMA Jl numer. Analysis* **4**, 169–184 (1984).
4. J. R. Dormand and P. J. Prince, A reconsideration of some embedded Runge–Kutta formulae. *J. Comput. appl. Math.* **15**(2), 203–211 (1986).
5. M. K. Horn, Fourth- and fifth-order scaled Runge–Kutta algorithms for treating dense output. *SIAM Jl numer. Analysis* **20**, 558–568 (1983).
6. L. F. Shampine, Some practical Runge–Kutta formulas. Report SAND84-0812 (1984).
7. L. F. Shampine, Interpolation for Runge–Kutta methods. *SIAM Jl numer. Analysis* **22**, 1014–1027 (1985).
8. W. H. Enright, K. R. Jackson, S. P. Norsett and P. G. Thomsen, Interpolants for Runge–Kutta formulas. Technical Report No. 180/85, Dept of Computer Science, Univ. of Toronto, Toronto, Ontario (1985).
9. I. Gladwell, L. F. Shampine, L. S. Baca and R. W. Brankin, Practical aspects of interpolation in Runge–Kutta codes. Numerical Analysis Report No. 102, Dept of Mathematics, Univ. of Manchester, Manchester, Lancs. (1985).

10. A. Ralston and P. Rabinowitz, *A First Course in Numerical Analysis*, 2nd edn. McGraw-Hill, London (1978).
11. J. C. Butcher, Coefficients for the study of Runge–Kutta integration processes. *J. Aust. math. Soc.* **3**, 185–201 (1963).
12. P. J. Prince and J. R. Dormand, High order embedded Runge–Kutta formulae. *J. Comput. appl. Math.* **7**(1), 67–75 (1981).
13. T. E. Hull, W. H. Enright, W. H. Fellen and A. E. Sedgwick, Comparing numerical methods for ordinary differential equations. *SIAM Jl numer. Analysis* **9**, 603–637 (1972).